



EPITECH INNOVATIVE PROJECT

PROJECT LOG DOCUMENT

PROMO 2025

Résumé du document

Le PLD (Project Log Document) est un document qui va accompagner notre projet tout au long de notre cursus EIP, et qui s'étoffera au fur et à mesure de l'avancement de celui-ci.

Il servira de support lors de nos rendez-vous mensuels avec notre référent EIP :

- Lors d'un *Kick Off*, il servira à définir le scope du sprint à venir ;
- Lors d'un *Follow Up*, il pourra être utilisé si nous souhaitons y apporter des modifications en cours de sprint (en accord avec notre référent EIP) ;
- Lors d'un *Delivery*, il servira de support à la validation du sprint passé.

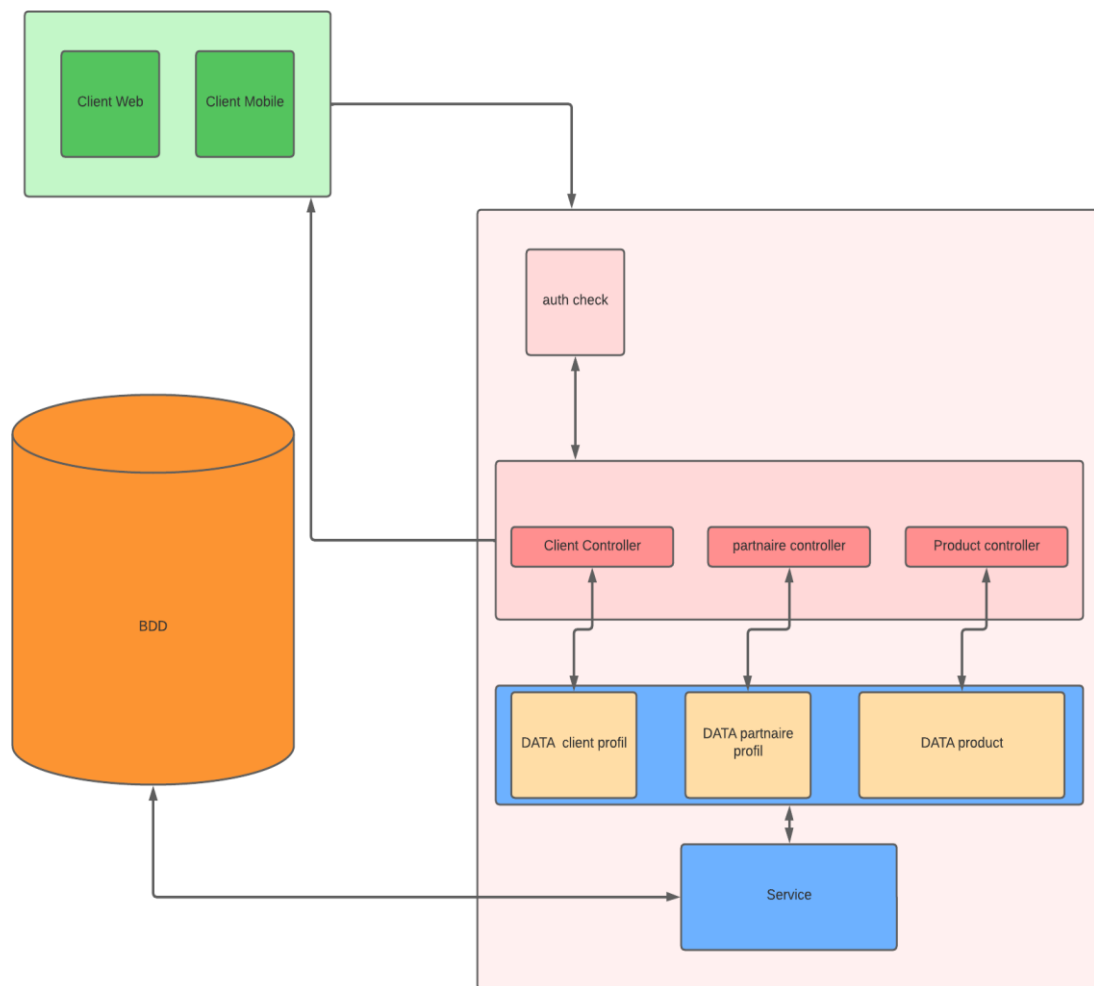
Plus généralement et sur le long terme, le PLD permet de conserver une trace de l'évolution et de l'avancement de notre projet au cours du temps.

En outre, lorsqu'il est bien préparé en amont ce document est une très bonne base pour tenir une réunion de projet, puisque tous les éléments principaux y sont abordés.

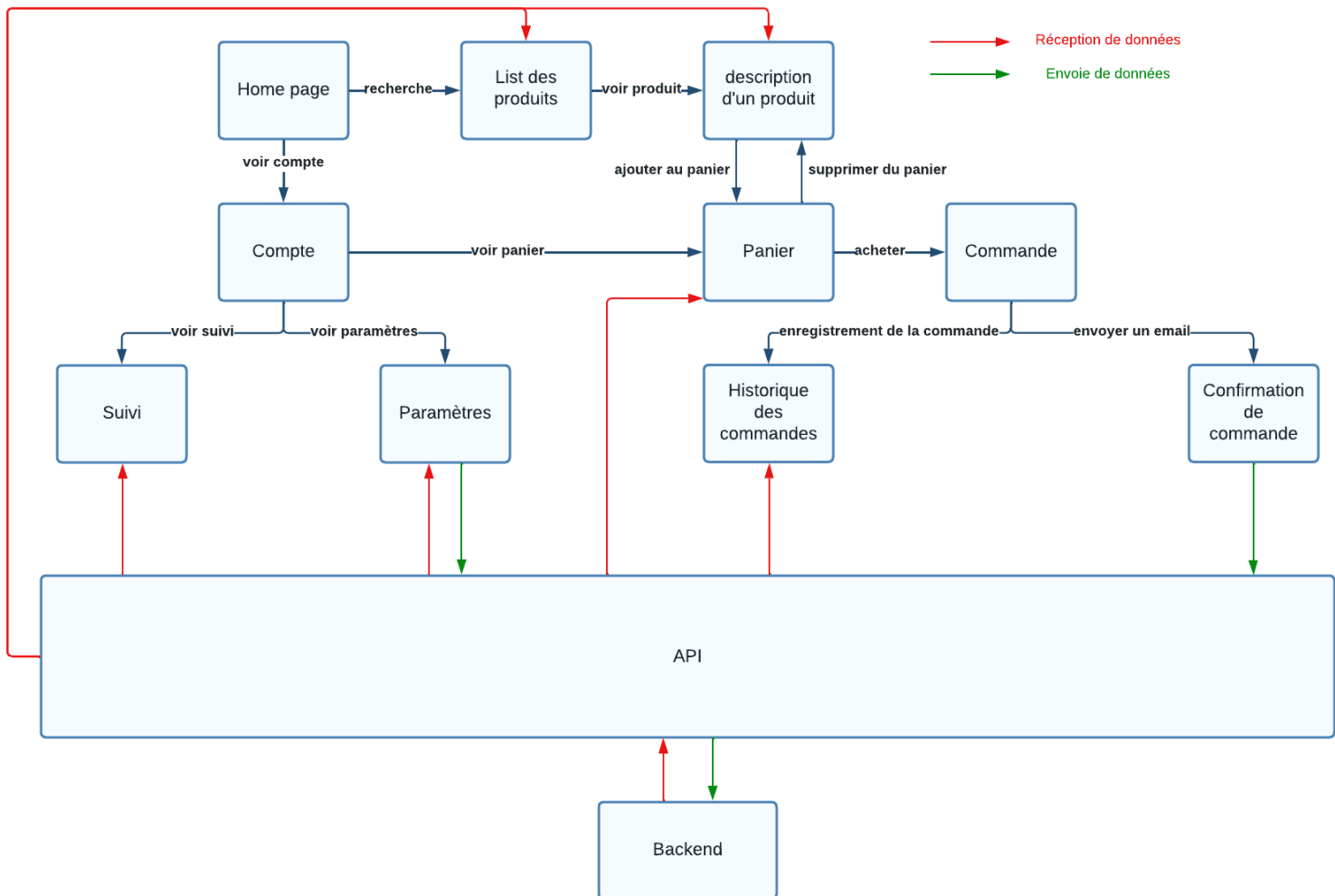
Sommaire

1. Shéma fonctionnel BackEnd	4
2. Shéma fonctionnel FrontEnd	5
3. Tableaux des stories BackEnd	6
4. Tableaux des stories FrontEnd	12
5. Tableaux des stories Mobile.....	19
6. Avancement.....	28

1.Schéma fonctionnel Backend



2.Schéma fonctionnel Frontend / Mobile



3. Tableaux des stories Backend

API			
1 Authentification	2 Shopping	3 Profil	4 Technique
1.1 Inscription Client	2.1 Récupération des articles disponibles	3.1 Informations Principales	4.1 Mise en place de la base de données
1.2 Connexion Client	2.2 Enregistrement d'un vêtement	3.2 Modification des informations principales	4.2 Dockeriser le projet
1.3 Inscription Entreprise	2.3 Calcul de l'éco-score	3.3 Calcul du score personnel	
1.4 Connexion Entreprise		3.4 calculs des données entreprise	

1.1 Inscription Client

En tant que

Je Veux

Client

M'inscrire

Description L'utilisateur de l'application doit pouvoir créer son compte sur la plateforme, afin de s'y connecter et d'utiliser pleinement l'application par la suite.

Definition Of Done :

- La route devra être en POST ;
- La route devra accepter un objet JSON dans le body de la requête ;
- L'objet JSON devra contenir une adresse e-mail, et un mot de passe ;
- En cas de succès, la route devra retourner un code HTTP 200 ;
- En cas d'erreur sur les données reçues, la route devra retourner un code HTTP 400 ;
- En cas d'erreur côté API, la route devra retourner un code HTTP 400 ;
- Les mots de passe devront être refusés s'ils font moins de huit caractères, ou s'ils ne contiennent pas des lettres et des chiffres ;
- Les adresses e-mails devront être uniques ;
- Les données en entrées devront être contrôlées afin de diminuer les risques de piratage ;
- La route devra être documentée ;
- Des tests unitaires pour chacun des cas devront être réalisés.

Charge Estimée

2 J/H

Personne assignée : Louis-Quentin

Progression (en %) : 100

1.2 Connexion Client

En tant que

Je Veux

Client

Me connecter

Description L'utilisateur de l'application doit pouvoir se connecter sur la plateforme, afin d'utiliser pleinement l'application par la suite.

Definition Of Done :

- La route devra être en POST ;
- La route devra accepter un objet JSON dans le body de la requête ;
- L'objet JSON devra contenir une adresse e-mail, et un mot de passe ;
- En cas de succès, la route devra retourner un code HTTP 200 ;
- En cas d'erreur sur les données reçues, la route devra retourner un code HTTP 400 ;
- En cas d'erreur côté API, la route devra retourner un code HTTP 400 ;
- Les mots de passe devront être refusés s'ils sont incorrects avec un code HTTP 400 ;
- Les emails devront être refusés s'ils sont incorrects avec un code HTTP 400 ;
- Les données en entrées devront être contrôlées afin de diminuer les risques de piratage ;
- En cas de réussite le client se verra attribuer un token jwt afin de réduire les risques de sécurité
- La route devra être documentée ;
- Des tests unitaires pour chacun des cas devront être réalisés.

Charge Estimée

3 J/H

Personne assignée : Louis-Quentin

Progression (en %) : 66

1.3 Inscription Entreprise

En tant que

Je Veux

Entreprise Partenaire

M'inscrire

Description Le partenaire doit pouvoir créer un compte pour son entreprise sur la plateforme, afin de s'y connecter et d'utiliser pleinement l'application par la suite, notamment enregistrer des articles à la vente et générer leur éco-score.

Definition Of Done :

- La route devra être en POST ;
- La route devra accepter un objet JSON dans le body de la requête ;
- L'objet JSON devra contenir une adresse e-mail, et un mot de passe ;
- En cas de succès, la route devra retourner un code HTTP 200 ;
- En cas d'erreur sur les données reçues, la route devra retourner un code HTTP 400 ;
- En cas d'erreur côté API, la route devra retourner un code HTTP 400 ;
- Les mots de passe devront être refusés s'ils font moins de huit caractères, ou s'ils ne contiennent pas des lettres et des chiffres ;
- Les adresses e-mails devront être uniques ;
- Les données en entrées devront être contrôlées afin de diminuer les risques de piratage ;
- La route devra être documentée ;
- Des tests unitaires pour chacun des cas devront être réalisés.

Charge Estimée

2 J/H

Personne assignée : Louis-Quentin

Progression (en %) : 100

1.4 Connexion Entreprise

En tant que

Je Veux

Entreprise partenaire	Me connecter
Description Le partenaire de Clothway doit pouvoir se connecter sur la plateforme, afin d'utiliser pleinement l'application par la suite.	
Definition Of Done : <ul style="list-style-type: none"> – La route devra être en POST ; – La route devra accepter un objet JSON dans le body de la requête ; – L'objet JSON devra contenir une adresse e-mail, et un mot de passe ; – En cas de succès, la route devra retourner un code HTTP 200 ; – En cas d'erreur sur les données reçues, la route devra retourner un code HTTP 400 ; – En cas d'erreur côté API, la route devra retourner un code HTTP 400 ; – Les mots de passe devront être refusés s'ils sont incorrects avec un code HTTP 400 ; – Les emails devront être refusés s'ils sont incorrects avec un code HTTP 400 ; – Les données en entrées devront être contrôlées afin de diminuer les risques de piratage ; – En cas de réussite le client se verra attribuer un token jwt afin de réduire les risques de sécurité – La route devra être documentée ; – Des tests unitaires pour chacun des cas devront être réalisés. 	
Charge Estimée	3 J/H
Personne assignée : Louis-Quentin	
Progression (en %) : 66	

2.1 Récupération des articles disponibles	
En tant que	Je Veux
Développeur	Récupérer la liste des articles disponibles
Description La liste de tous les vêtements disponibles doit être récupérée afin de tenir à jour les produits disponibles ou non sur le site.	
Definition Of Done : <ul style="list-style-type: none"> – La route devra être en GET ; – En cas de succès, la route devra retourner un code HTTP 200 ; – En cas d'erreur sur les données reçues, la route devra retourner un code HTTP 400 ; – En cas d'erreur côté API, la route devra retourner un code HTTP 400 ; – La route devra être documentée ; – Gestion des filtres. – Gestion des préférences personnelles de l'utilisateur. – Des tests unitaires pour chacun des cas devront être réalisés. 	
Charge Estimée	1J/H
Personne assignée : Mattéo	
Progression (en %) : 100	

2.2 Enregistrement d'un vêtement	
En tant que	Je Veux
Entreprise	Enregistrer un vêtement
Description L'entreprise doit pouvoir rentrer les différentes informations de son produit dans l'objectif de le mettre en ligne sur le site.	

Definition Of Done :

- La route devra être en POST ;
- La route devra accepter un objet JSON dans le body de la requête ;
- L'objet JSON devra contenir les informations du produit (noms, marque, prix, catégorie, image, pays origine) ;
- Gestion des inputs rentrés par l'entreprise.
- Vérifications des données.
- En cas de succès, la route devra retourner un code HTTP 200 ;
- En cas d'erreur sur les données reçues, la route devra retourner un code HTTP 400 ;
- En cas d'erreur côté API, la route devra retourner un code HTTP 400 ;
- La route devra être documentée ;
- Des tests unitaires pour chacun des cas devront être réalisés.

Charge Estimée**2 J/H****Personne assignée : Mattéo****Progression (en %) : 100****2.3 Calcul de l'éco-score****En tant que****Je Veux****Développeur****Calculer l'éco-score d'un produit**

Description L'éco-score va être une note (entre 1 et 5) qui va définir si le produit est éco-responsable ou non. Ce score va être calculé avec les données reçues lors de l'enregistrement d'un produit.

Definition Of Done :

- La route devra être en POST ;
- Les informations enregistrées du produit serviront dans le calcul du produit ;
- En cas de succès, la route devra retourner un code HTTP 200 ;
- En cas d'erreur sur les données reçues, la route devra retourner un code HTTP 400 ;
- Mise en place du calcul (critères -> pays origine, matériaux utilisés, type du produit).
- En cas d'erreur côté API, la route devra retourner un code HTTP 400 ;
- La route devra être documentée ;
- Des tests unitaires pour chacun des cas devront être réalisés.

Charge Estimée**2 J/H****Personne assignée : Mattéo****Progression (en %) : 100****4.1 Mise en place de la base de données****En tant que****Je Veux****Développeur****Mettre en place la base de données**

Description La base de données PostgreSQL doit être déployée, afin que l'API puisse accéder aux données.

Definition Of Done :

- La version Community Server devra être déployée ;
- La version de PostgreSQL devra être la 15 ou ultérieure ;

- La version devra être stable ;
- La base de données ne devra écouter qu'en local sur le port 8080 ;
- L'authentification devra être activée ;
- Un compte administrateur devra être créé pour l'administrateur de la base ;

Charge Estimée	0.5 J/H
----------------	---------

Personne assignée : Louis-Quentin	
-----------------------------------	--

Progression (en %) : 0	
------------------------	--

4.2 Dockeriser le projet

En tant que	Je Veux
-------------	---------

Développeur	Dockeriser le projet
-------------	----------------------

Description Le projet doit être contenu dans un docker-compose

Definition Of Done :

- Un Dockerfile pour la partie Back-End qui va installer les dépendances (Golang) ;
- Un Dockerfile pour la partie Front-End qui va installer les dépendances (reactJS) ;
- Un Dockerfile pour la partie Mobile qui va installer les dépendances (Flutter) ;
- Un DockerCompose qui va contenir les 3 docker.
- Réaliser une partie dans le DockerCompose dédiée à la base de données.
- Documenter les Dockerfile.
- Documenter le DockerCompose.

Charge Estimée	3 J/H
----------------	-------

Personne assignée : Mattéo	
----------------------------	--

Progression (en %) : 100	
--------------------------	--

3.1 Informations principales

En tant que	Je Veux
-------------	---------

Client	Accéder aux données essentielles de mon profil
--------	--

Description Le client de Clothway doit pouvoir charger les données de bases de son profil sur la plateforme, ainsi que les modifier si nécessaire.

Definition Of Done :

- La route pour charger le profil devra être en GET ;
- La route devra accepter un token jwt dans la requête ;
- La route devra utiliser le token présent pour authentifier l'utilisateur afin d'éviter de retourner le mauvais profil
- En cas de succès, la route devra retourner un code HTTP 200 ;
- En cas d'erreur sur les données reçues, la route devra retourner un code HTTP 400 ;
- En cas d'erreur côté API, la route devra retourner un code HTTP 400 ;
- En cas de token invalide l'utilisateur sera redirigé sur la page d'authentification ;
- Les données en entrées devront être contrôlées afin de diminuer les risques de piratage ;
- En cas de réussite le client se verra retourner les données de son profil
- La route devra être documentée ;
- Des tests unitaires pour chacun des cas devront être réalisés.

Charge Estimée	2 J/H
----------------	-------

Personne assignée : Louis-Quentin	
-----------------------------------	--

Progression (en %) : 0	
------------------------	--

3.2 Modification des informations principales

En tant que	Je Veux
Client	Modifier les données essentielles de mon profil
Description Le client de Clothway doit pouvoir modifier les données de bases de son profil sur la plateforme.	
Definition Of Done : <ul style="list-style-type: none"> – La route pour modifier le profil devra être en POST ; – La route devra accepter un token jwt dans la requête ; – La route devra utiliser le token présent pour authentifier l'utilisateur afin d'éviter de retourner le mauvais profil – La route devra accepter un objet JSON dans le body de la requête ; – L'objet JSON devra contenir le champ à modifier ; – En cas de succès, la route devra retourner un code HTTP 200 ; – En cas d'erreur sur les données reçues, la route devra retourner un code HTTP 400 ; – En cas d'erreur côté API, la route devra retourner un code HTTP 400 ; – En cas de token invalide l'utilisateur sera redirigé sur la page d'authentification ; – Les données en entrées devront être contrôlées afin de diminuer les risques de piratage ; – En cas de réussite le client se verra retourner les données de son profil actualisé ; – La route devra être documentée ; – Des tests unitaires pour chacun des cas devront être réalisés. 	
Charge Estimée	2 J/H
Personne assignée : Mattéo	
Progression (en %) : 0	

3.3 Calcul du score personnel

En tant que	Je Veux
Client	Consulter mon éco-score
Description Le client de Clothway doit pouvoir consulter son éco-score sur son profil	
Definition Of Done : <ul style="list-style-type: none"> – La route devra être en GET ; – La route devra accepter un token jwt dans la requête ; – La route devra utiliser le token présent pour authentifier l'utilisateur afin d'éviter de retourner le mauvais profil <p>Tous les articles achetés au préalable par l'utilisateur seront utilisés afin de définir l'éco-score à attribuer au client</p> <ul style="list-style-type: none"> – En cas de succès, la route devra retourner un code HTTP 200 ; – En cas d'erreur sur les données reçues, la route devra retourner un code HTTP 400 ; – En cas d'erreur côté API, la route devra retourner un code HTTP 400 ; – En cas de token invalide l'utilisateur sera redirigé sur la page d'authentification ; – Les données en entrées devront être contrôlées afin de diminuer les risques de piratage ; – En cas de réussite le client se verra retourner les données de son profil actualisé ; – La route devra être documentée ; – Des tests unitaires pour chacun des cas devront être réalisés. 	
Charge Estimée	2 J/H

Personne assignée : Mattéo

Progression (en %) : 0

4. Tableaux des stories Frontend

Front web			
1 Authentification	2 Page d'accueil	3 Toolbar	4 Produits
1.1 Inscription Client	2.1 Bannière d'accueil	3.1 Burger Menu	4.1 Carte produit
1.2 Connexion Client	2.2 Bandeau d'actualité	3.2 Bouton connexion 3.3 Bouton de compte	4.2 Shop list
1.3 Inscription Entreprise	2.3 Bandeau réseaux, confidentialités, informations		
1.4 Connexion Entreprise			

1.1 Inscription Client

En tant que

Je Veux

Client

M'inscrire sur le site

Description L'utilisateur pourra s'inscrire sur notre site afin de pouvoir l'utiliser par la suite.

Definition Of Done :

- La route d'inscription doit être une requête POST.
- La route doit accepter un objet JSON dans le corps de la requête.
- L'objet JSON doit contenir une adresse e-mail et un mot de passe.
- Si l'inscription est réussie, la route doit renvoyer un code de statut HTTP 200.
- En cas d'erreur dans les données reçues, la route doit renvoyer un code de statut HTTP 400.
- En cas d'erreur côté API, la route doit renvoyer un code de statut HTTP 400.
- Les mots de passe doivent être rejetés s'ils font moins de huit caractères ou ne contiennent pas à la fois des lettres et des chiffres.
- Les adresses e-mail doivent être uniques.
- Pour mettre en œuvre ce formulaire d'inscription, un composant React devrait être créé qui permet à l'utilisateur d'entrer son adresse e-mail et son mot de passe. Le composant devrait gérer la validation des données d'entrée et envoyer une requête POST à la route API pour l'inscription. La réponse de l'API doit être traitée et affichée à l'utilisateur en

conséquence.

Charge Estimée

2 J/H

Personne assignée : Mathis

Progression (en %) : 100

1.2 Connexion Client

En tant que

Je Veux

Client

Accéder aux différents types de produits

Description L'utilisateur pourra se connecter sur le site afin de pouvoir y accéder.**Definition Of Done :**

- La route de connexion doit être une requête POST.
- La route doit accepter un objet JSON dans le corps de la requête.
- L'objet JSON doit contenir une adresse e-mail et un mot de passe.
- Si la connexion est réussie, la route doit renvoyer un code de statut HTTP 200.
- Si l'adresse e-mail n'existe pas, la route doit renvoyer un code de statut HTTP 400.
- Si le mot de passe est incorrect, la route doit renvoyer un code de statut HTTP 400.
- En cas d'erreur côté API, la route doit renvoyer un code de statut HTTP 400.
- Les données d'entrée doivent être validées pour réduire le risque de piratage.
- La route doit être documentée.
- Pour mettre en œuvre ce formulaire de connexion, un composant React devrait être créé qui permet à l'utilisateur d'entrer son adresse e-mail et son mot de passe. Le composant devrait gérer la validation des données d'entrée et envoyer une requête POST à la route API pour la connexion. La réponse de l'API doit être traitée et affichée à l'utilisateur en conséquence.

Charge Estimée

2 J/H

Personne assignée : Mathis

Progression (en %) : 100

1.3 Inscription Entreprise

En tant que

Je Veux

Entreprise

M'inscrire sur le site

Description L'entreprise pourra s'inscrire sur notre site afin de pouvoir l'utiliser par la suite.**Definition Of Done :**

- Avoir un bouton cliquable
- La route d'inscription doit être une requête POST.
- La route doit accepter un objet JSON dans le corps de la requête.
- L'objet JSON doit contenir une adresse e-mail et un mot de passe.
- Si l'inscription est réussie, la route doit renvoyer un code de statut HTTP 200.
- En cas d'erreur dans les données reçues, la route doit renvoyer un code de statut HTTP 400.

- En cas d'erreur côté API, la route doit renvoyer un code de statut HTTP 400.
- Les mots de passe doivent être rejetés s'ils font moins de huit caractères ou ne contiennent pas à la fois des lettres et des chiffres.
- Les adresses e-mail doivent être uniques.
- Des tests unitaires pour chaque cas doivent être effectués.
- Pour mettre en œuvre ce formulaire d'inscription, un composant React devrait être créé qui permet à l'entreprise d'entrer son adresse e-mail et son mot de passe. Le composant devrait gérer la validation des données d'entrée et envoyer une requête POST à la route API pour l'inscription. La réponse de l'API doit être traitée et affichée à l'utilisateur en conséquence.

Charge Estimée	2 J/H
Personne assignée : Steeven	
Progression (en %) : 100	

1.4 Connexion Entreprise	
En tant que	Je Veux
Entreprise	Me connecter sur le site
Description L'entreprise pourra se connecter sur le site afin de pouvoir y accéder	
Definition Of Done : <ul style="list-style-type: none"> • La route de connexion doit être une requête POST. • La route doit accepter un objet JSON dans le corps de la requête. • L'objet JSON doit contenir une adresse e-mail et un mot de passe. • Si la connexion est réussie, la route doit renvoyer un code de statut HTTP 200. • Si l'adresse e-mail n'existe pas, la route doit renvoyer un code de statut HTTP 400. • Si le mot de passe est incorrect, la route doit renvoyer un code de statut HTTP 400. • En cas d'erreur côté API, la route doit renvoyer un code de statut HTTP 400. • Les données d'entrée doivent être validées pour réduire le risque de piratage. • La route doit être documentée. • Des tests unitaires pour chaque cas doivent être effectués. • Pour mettre en œuvre ce formulaire de connexion, un composant React devrait être créé qui permet à l'entreprise d'entrer son adresse e-mail et son mot de passe. Le composant devrait gérer la validation des données d'entrée et envoyer une requête POST à la route API pour l'inscription. La réponse de l'API doit être traitée et affichée à l'utilisateur en conséquence. 	
Charge Estimée	2 J/H
Personne assignée : Steeven	
Progression (en %) : 100	

2.1 Bannière d'accueil	
En tant que	Je Veux
Client	Accéder à la bannière d'accueil
Description La bannière d'accueil sera la première page qu'une personne verra lorsqu'elle accèdera à notre site	

Definition Of Done :

- Créer un composant React pour la bannière d'accueil qui sera affichée en haut de la page d'accueil. Ce composant devrait inclure des images et du texte pour promouvoir les produits phares de votre site.
- Le composant doit être conçu pour être réactif et s'adapter à différentes tailles d'écran pour garantir une expérience utilisateur cohérente sur tous les appareils.
- Pour améliorer l'expérience utilisateur, vous pouvez ajouter des animations pour attirer l'attention de l'utilisateur sur les produits en vedette.
- La bannière d'accueil doit être liée aux pages pertinentes de votre site afin que les utilisateurs puissent accéder rapidement aux produits qui les intéressent.
- Il est important de valider les données utilisées pour créer la bannière d'accueil et de gérer les erreurs en affichant des messages d'erreur appropriés à l'utilisateur en cas de problèmes de réseau ou de données invalides.
- Pour améliorer la performance de votre site, vous pouvez utiliser des techniques de chargement asynchrone pour charger les images et le contenu de la bannière d'accueil en arrière-plan.
- Des tests unitaires doivent être effectués pour s'assurer que chaque fonctionnalité de la bannière d'accueil fonctionne correctement. Ces tests doivent être exécutés à chaque modification de code pour s'assurer que tout fonctionne correctement.

Charge Estimée**1 J/H****Personne assignée : Steeven****Progression (en %) : 0****2.2 Bandeau d'actualité****En tant que****Je Veux****Client****Accéder le bandeau d'actualité**

Description Le bandeau d'actualité sera un petit élément de notre site qui montrera des informations récentes concernant l'écologie et la planète.

Definition Of Done :

- Créer un composant React pour le bandeau d'actualité qui affichera les dernières actualités de votre site de vêtements. Le composant devrait inclure une méthode pour récupérer les données d'actualités à partir de l'API.
- Une fois que les données sont récupérées, le composant doit afficher les dernières actualités sous forme de liste déroulante ou de carrousel selon le design choisi. Pour cela, il est préférable d'utiliser une bibliothèque de composants React telle que React Bootstrap ou Material-UI pour accélérer le processus de développement.

- Le bandeau d'actualité doit être dynamique et actualisé en temps réel. Vous pouvez utiliser des bibliothèques telles que React Spring ou React-Flip-Move pour ajouter des animations pour une meilleure expérience utilisateur.
- Il est important de valider les données renvoyées par l'API et de gérer les erreurs en affichant des messages d'erreur appropriés à l'utilisateur en cas de problèmes de réseau ou de données invalides.
- Des tests unitaires doivent être effectués pour s'assurer que chaque fonctionnalité du bandeau d'actualité fonctionne correctement. Ces tests doivent être exécutés à chaque modification de code pour s'assurer que tout fonctionne correctement.

Charge Estimée	3.5 J/H
Personne assignée : Steeven	
Progression (en %) 100	

2.3 Bandeau réseaux, confidentialités	
En tant que	Je Veux
Client	Accéder aux différents types de produits
Description Le bandeau réseaux, confidentialités sera la promotion de nos réseaux sociaux sur notre site ainsi que les confidentialités de celui si	
Definition Of Done : <ul style="list-style-type: none"> • Pour créer le bandeau réseaux/confidentialité en React pour l'inscription sur un site de vêtements. • Il faut créer un composant qui permet à l'utilisateur d'entrer son adresse e-mail et son mot de passe. • Valider les données d'entrée et envoyer une requête POST à la route API pour l'inscription. • Le composant doit gérer la réponse de l'API en conséquence, avec des messages de confirmation ou d'erreur appropriés. • Des tests unitaires doivent être effectués pour chaque cas de la requête POST, le code doit être documenté et respecter les normes de sécurité. • Un lien vers la page de politique de confidentialité du site doit être inclus dans le bandeau. 	
Charge Estimée	0.5 J/H
Personne assignée : Steeven	
Progression (en %) : 100	

3.1 Burger Menu	
En tant que	Je Veux
Client	Accéder aux différents types de produits
Description L'utilisateur doit pouvoir cliquer sur une icône pour voir et choisir le type de produit qu'il souhaite rechercher.	
Definition Of Done : <ul style="list-style-type: none"> • Avoir un bouton cliquable 	

- Avoir une icône pour le bouton
- Ouvrir une page de dialogue sur le coté
- Avoir toutes mes catégories et sous catégories des produits
- Diriger vers la page du type du produit quand in clique
- Fermer la page de dialogue quand on clique dans le vide
- Faire en sorte que la page de dialogue soit « responsive »
- Faire en sorte que la page de dialogue soit esthétique et facile d'utilisation
- Bien proportionner le l'icône et placer le bouton

Charge Estimée	2.5 J/H
----------------	---------

Personne assignée : Mathis	
----------------------------	--

Progression (en %) : 100	
--------------------------	--

3.4 Bouton de connexion

En tant que	Je Veux
-------------	---------

Client	Aller sur la page de connexion
--------	--------------------------------

Description L'utilisateur doit pouvoir cliquer sur une icône pour aller à la page de connexion.

Definition Of Done :

- Avoir un bouton cliquable
- Avoir une icône comme bouton
- Revenir sur la page principale
- Bien proportionner le logo et placer le bouton
- Faire en sorte le bouton soit « responsive »

Charge Estimée	0.5 J/H
----------------	---------

Personne assignée : Mathis	
----------------------------	--

Progression (en %) : 0	
------------------------	--

3.5 Bouton de compte

En tant que	Je Veux
-------------	---------

Client	Aller sur mon compte
--------	----------------------

Description L'utilisateur doit pouvoir cliquer sur une icône pour aller à la page de son compte.

Definition Of Done :

- Avoir un bouton cliquable
- Avoir une icône ou sa photo comme bouton
- Rediriger vers la page du compte de l'utilisateur
- Bien proportionner le logo et placer le bouton
- Faire en sorte le bouton soit « responsive »

Charge Estimée	0.5 J/H
----------------	---------

Personne assignée : Mathis	
----------------------------	--

Progression (en %) : 0	
------------------------	--

4.1 Carte Produit

En tant que	Je Veux
Client	Accéder à la carte produit d'un vêtement
Description La carte produit correspond à la description du produit accompagné d'une photo et de son éco-score	
Definition Of Done : <ul style="list-style-type: none"> Avoir un bouton cliquable Concevoir la structure du composant : Le composant doit être conçu pour afficher les détails du produit, y compris l'image, le nom, la description, le prix, la disponibilité et les options de taille et de couleur. Le composant peut également inclure des boutons pour ajouter le produit au panier ou pour voir plus de détails. Gérer les interactions utilisateur : Le composant doit être capable de gérer les interactions utilisateur telles que la sélection de la taille et de la couleur, l'ajout au panier et la visualisation des détails du produit. Récupérer les données du produit : Les données du produit peuvent être récupérées à partir d'une source de données, telles qu'une API de produits, et affichées dans le composant. Le site doit aussi récupérer l'éco score du vêtement à partir des données renvoyées par l'api Gérer la validation de la sélection des options : Le composant doit être capable de gérer la validation de la sélection de la taille et de la couleur en fonction de la disponibilité du produit. Gérer l'ajout au panier : Le composant doit être capable de gérer l'ajout du produit au panier en envoyant une requête POST à l'API de panier avec les détails du produit. Gérer les erreurs : Le composant doit être capable de gérer les erreurs qui peuvent se produire lors de l'ajout du produit au panier, telles qu'une erreur de serveur ou une erreur de validation. Tester le composant : Des tests unitaires doivent être effectués pour chaque cas de sélection des options, d'ajout au panier et d'affichage des détails du produit. 	
Charge Estimée	3.5 J/H
Personne assignée : Mathis	
Progression (en %) : 0	

4.2 Shop list

En tant que	Je Veux
Client	Voir les différents produits sur une page
Description L'utilisateur doit pouvoir sélectionner des filtres pour afficher les produits dans la ou les catégories qu'il souhaite.	
Definition Of Done : <ul style="list-style-type: none"> Récupérer les produits via une requête GET Faire une search bar Prendre en compte les filtres Afficher les produits dans des cartes Mettre toutes les données du produit dans la carte 	

- Proposer des produits grâce aux cookies
- Bien proportionner et placer les éléments
- Faire en sorte que les éléments soient « responsive »

Charge Estimée

4 J/H

Personne assignée : Mathis

Progression (en %) : 12.5

5. Tableaux des stories Mobile

1 Authentification	2 Page d'accueil	3 Toolbar	4 Planification
1.1 Inscription Client	2.1 Bannière d'accueil	3.1 Menu déroulant	4.1 Maquette du projet
1.2 Connexion Client	2.2 Bandeau réseaux, confidentialités, informations	3.2 Logo pour revenir à la HomePage	4.2 Schéma fonctionnel
1.3 Mot de passe oublié		3.3 Page de compte	

1.1 Inscription Client

En tant que

Je Veux

Client

M'inscrire sur le site

Description L'utilisateur pourra s'inscrire sur notre site afin de pouvoir l'utiliser par la suite.

Definition Of Done :

- Avoir un bouton cliquable avec un texte "inscription".
- Diriger vers la page d'inscription quand il clique.
- La route d'inscription doit être une requête POST.
- La route doit accepter un objet JSON dans le corps de la requête.
- Pour mettre en œuvre ce formulaire d'inscription, l'utilisateur devrait pouvoir entrer son adresse e-mail et son mot de passe.
- Gérer la validation des données d'entrée et envoyer une requête POST à la route API pour l'inscription. La réponse de l'API doit être traitée et affichée à l'utilisateur en conséquence.
- L'objet JSON doit contenir une adresse e-mail et un mot de passe.
- Si l'inscription est réussie, la route doit renvoyer un code de statut HTTP 200.
- En cas d'erreur dans les données reçues, la route doit renvoyer un code de statut HTTP 400.
- En cas d'erreur côté API, la route doit renvoyer un code de statut HTTP 400.

- Les mots de passe doivent être rejetés s'ils font moins de huit caractères ou ne contiennent pas à la fois des lettres et des chiffres.
- Les adresses e-mail doivent être uniques.
- La route doit être documentée.

Charge Estimée	2,5 J/H
----------------	---------

Personne assignée : Alexis	
----------------------------	--

Progression (en %) : 100	
--------------------------	--

1.2 Connexion Client

En tant que	Je Veux
-------------	---------

Client	Accéder aux différents types de produits
--------	--

Description L'utilisateur pourra se connecter sur le site afin de pouvoir y accéder.

Definition Of Done :

- Avoir un bouton cliquable avec un texte "connexion" dessus.
- Diriger vers la page de connexion quand il clique.
- La route de connexion doit être une requête POST.
- La route doit accepter un objet JSON dans le corps de la requête.
- Pour mettre en œuvre ce formulaire d'inscription, l'utilisateur devrait pouvoir entrer son adresse e-mail et son mot de passe.
- Gérer la validation des données d'entrée et envoyer une requête POST à la route API pour la connexion. La réponse de l'API doit être traitée et affichée à l'utilisateur en conséquence.
- L'objet JSON doit contenir une adresse e-mail et un mot de passe.
- Si la connexion est réussie, la route doit renvoyer un code de statut HTTP 200.
- Si l'adresse e-mail n'existe pas, la route doit renvoyer un code de statut HTTP 400.
- Si le mot de passe est incorrect, la route doit renvoyer un code de statut HTTP 400.
- En cas d'erreur côté API, la route doit renvoyer un code de statut HTTP 400.
- Les données d'entrée doivent être validées pour réduire le risque de piratage.
- La route doit être documentée.

Charge Estimée	2,5 J/H
----------------	---------

Personne assignée : Redouane	
------------------------------	--

Progression (en %) : 100	
--------------------------	--

1.3 Mot de passe oublié

En tant que	Je Veux
-------------	---------

Client	Accéder aux différents types de produits
--------	--

Description L'utilisateur pourra se connecter sur le site afin de pouvoir y accéder.

Definition Of Done :

- Avoir un bouton cliquable avec un texte "Mot de passe oublié" dessus.
- Diriger vers la page d'oubli de mot de passe quand il clique.
- La route de mot de passe oublié doit être une requête POST.
- La route doit accepter un objet JSON dans le corps de la requête.
- Pour permettre de se connecter malgré son mot de passe oublié, l'utilisateur devra entrer son adresse e-mail pour recevoir un mot de passe provisoire.

- Gérer la validation des données d'entrée et envoyer une requête POST à la route API pour le mot de passe oublié. La réponse de l'API doit être traitée et affichée à l'utilisateur en conséquence.
- L'objet JSON doit contenir une adresse e-mail.
- Si le mot de passe provisoire est envoyé par mail, la route doit renvoyer un code de statut HTTP 200.
- Si l'adresse e-mail n'existe pas, la route doit renvoyer un code de statut HTTP 400.
- En cas d'erreur côté API, la route doit renvoyer un code de statut HTTP 400.
- La route doit être documentée.
- Diriger vers la page d'entrer du mot de passe provisoire si la route nous envoie le code de statut HTTP 200.
- Gérer la validation des données d'entrée et envoyer une requête POST à la route API pour le mot de passe provisoire. La réponse de l'API doit être traitée et affichée à l'utilisateur en conséquence.
- Si le mot de passe provisoire est incorrect, la route doit renvoyer un code de statut HTTP 400.
- Si le mot de passe est correct, la route doit renvoyer un code de statut HTTP 200.
- Si le code de statut HTTP est 200, rediriger le client vers une page de création d'un nouveau mot de passe contenant au minimum 8 caractères, et à la fois des chiffres et des lettres.
- En cas d'erreur dans les données reçues du nouveau mot de passe, la route doit renvoyer un code de statut HTTP 400.

Charge Estimée	3 J/H
Personne assignée : Alexis	
Progression (en %) : 0	

2.1 Bannière d'accueil	
En tant que	Je Veux
Client	Accéder à la bannière d'accueil
Description La bannière d'accueil sera la première page qu'une personne verra lorsqu'elle accèdera à notre application	
Definition Of Done : <ul style="list-style-type: none"> • Créer une bannière d'accueil qui sera affichée en haut de la page d'accueil. Cette bannière devra inclure des images et du texte pour promouvoir les produits phares de l'application. • La bannière d'accueil doit être liée aux pages pertinentes de l'application afin que les utilisateurs puissent accéder rapidement aux produits qui les intéressent. • Il est important de valider les données utilisées pour créer la bannière d'accueil et de gérer les erreurs en affichant des messages d'erreur appropriés à l'utilisateur en cas de problèmes de réseau ou de données invalides. 	
Charge Estimée	2 J/H
Personne assignée : Alexis	
Progression (en %) : 100	

4.1 Maquette du projet

En tant que	Je Veux
Développeur	Établir le plan de mon projet
Description : la maquette du projet doit être réalisée pour pouvoir ensuite produire l'application mobile à l'identique	
Definition Of Done : <ul style="list-style-type: none"> Faire la page d'introduction(login, register, mdp oublié) Faire la HomePage -> comprenant le bandeau d'actualité & bandeau d'accueil & la barre de recherche & les produits "pour vous" & les différentes catégories de vêtements & bandeau réseaux/confidentialité & aperçu de la sholist Faire le Menu déroulant voir tâche 3.1 pour le détail Faire la page de compte voir tâche 3.3 Faire la page de settings Faire la page panier Faire l'affichage de l'éco score d'un produit 	
Charge Estimée	4 J/H
Personne assignée : Alexis	
Progression (en %) : 100	

4.2 Schéma fonctionnel

En tant que	Je Veux
Développeur	Accéder au plan de mon application
Description : Pouvoir identifier rapidement et simplement les parties à fournir afin d'avancer sur les fonctionnalités	
Definition Of Done : <ul style="list-style-type: none"> Apprendre à utiliser LucidChart afin de générer les diagrammes les plus réaliste possible Chaque partie du Figma de la tâche 4.1 doit apparaitre distinctement et dans le détail sur un diagramme lucidchart 	
Charge Estimée	1.5 J/H
Personne assignée : Redouane	
Progression (en %) : 33	

2.2 Bandeau réseaux, confidentialités

En tant que	Je Veux
Client	Accéder aux différents types de produits
Description Le bandeau réseaux, confidentialités sera la promotion de nos réseaux sociaux sur notre site ainsi que les confidentialités de celui si	
Definition Of Done : <ul style="list-style-type: none"> • Créer le bandeau réseaux/confidentialité. • Un lien vers la page de politique de confidentialité de l'Application mobile doit être inclus dans le bandeau. • Créer une page de politique de confidentialité de l'Application mobile. • Avoir plusieurs boutons cliquables avec un logo associé à un de nos réseaux sociaux (exemple : instagram). • Envoyer une requête POST à la route API lorsque l'on clique sur un des boutons associés. • Redirection vers nos réseaux sociaux lorsque l'on reçoit la réponse HTTP 200. • Un lien vers la page de politique de confidentialité de l'Application mobile doit être inclus dans le bandeau. • Créer une page de politique de confidentialité de l'Application mobile. • Avoir un bouton cliquable avec le logo de confidentialité. • Redirection vers la page de politique de confidentialité lorsque l'on clique sur le bouton. • Le code doit être documenté et respecter les normes de sécurité. 	
Charge Estimée	3 J/H
Personne assignée : Redouane	
Progression (en %) : 0	

3.1 Menu déroulant

En tant que	Je Veux
Client	Accéder aux différents types de produits
Description L'utilisateur doit pouvoir cliquer sur une icône pour voir et choisir le type de produit qu'il souhaite rechercher.	
Definition Of Done : <ul style="list-style-type: none"> • Avoir un bouton cliquable • Avoir une icône pour le bouton • Ouvrir une page de dialogue sur le coté • Avoir toutes les fonctionnalités utilisateurs • Diriger vers la page du type du produit quand il clique • Fermer la page de dialogue quand on clique dans le vide • Faire en sorte que la page de dialogue soit « responsive » • Faire en sorte que la page de dialogue soit esthétique et facile d'utilisation • Bien proportionner le l'icône et placer le bouton 	
Charge Estimée	2 J/H
Personne assignée : Redouane	
Progression (en %) : 100	

3.2 Logo pour revenir à la HomePage

En tant que	Je Veux
Client	Revenir sur la page principale
Description L'utilisateur doit pouvoir cliquer sur une icône pour revenir sur la page principale.	
Definition Of Done : <ul style="list-style-type: none"> Avoir un bouton cliquable Avoir le logo de ClothWay pour bouton Revenir sur la page principale Bien proportionner le logo et placer le bouton Faire en sorte le bouton soit « responsive » 	
Charge Estimée	0.5 J/H
Personne assignée : Alexis	
Progression (en %) : 0	

3.3 Page de compte

En tant que	Je Veux
Client	Aller sur mon compte
Description L'utilisateur doit pouvoir cliquer sur une icône pour aller à la page de son compte.	
Definition Of Done : <ul style="list-style-type: none"> Avoir un bouton cliquable Avoir une icône ou sa photo comme bouton Rediriger vers la page du compte de l'utilisateur Bien proportionner le logo et placer le bouton Faire en sorte le bouton soit « responsive » Créer la page du compte Personnaliser son avatar Retrouver toutes ses statistiques "vulgarisées" 	
Charge Estimée	3 J/H
Personne assignée : Redouane	
Progression (en %) : 100	

6. Avancement

Avancement global pour ce rendez-vous
Sur quoi avez-vous collectivement avancé depuis le dernier RDV ? Quel est le % d'avancement de chacune des parties de votre projet
<ul style="list-style-type: none"> FrontEnd : /bannière d'accueil /bandeau d'actualité / Burger Menu /Bouton de connexion / bouton de compte / ShopList-> 100 %. BackEnd : connexion entreprise / inscription entreprise / calcul de l'éco-score / enregistrement d'un vêtement-> 100 %. Mobile : maquette figma / page de compte schéma / schéma fonctionnel -> 100 %

	Travail (liste des tâches détaillées finies ou en cours)
Mathis	Taches terminées ou en cours: <ul style="list-style-type: none"> • Inscription client • Connexion client • Burger menu • Bouton de connexion • Bouton de compte • Shop List
Steeven	Taches terminées ou en cours: <ul style="list-style-type: none"> • Inscription entreprise • Connexion entreprise • Bannière d'accueil • Bandeau d'actualité
Louis-Quentin	Taches terminées ou en cours: <ul style="list-style-type: none"> • Inscription client • Connexion client • Inscription entreprise • Connexion entreprise
Mattéo	Taches terminées ou en cours: <ul style="list-style-type: none"> • Dockeriser le projet • Récupération de tous les articles • Enregistrement d'un article • Calcul de l'éco score
Alexis	Taches terminées ou en cours: <ul style="list-style-type: none"> • Inscription client • Bannière d'accueil • Maquette figma
Redouane	Taches terminées ou en cours: <ul style="list-style-type: none"> • Connexion utilisateur • Menu déroulant • Schéma fonctionnel • Page de compte
Points bloquants	
Rappel des points bloquants et tickets en souffrance	
Commentaire général	
Commentaire libre et questions	